**Mishal Ch**

**BSEF21M522**

**Project Documentation**

**Video Search Engine**

---

All video, audio, text files are included in zip file which was generated by this program.

**Load all folder in Vs code**

**1. Introduction:**

**The Speech to Text Converter with Timestamps is a Python program that extracts speech from audio and video files, converts it into text, and provides timestamps for individual words. This project aims to facilitate the conversion of spoken content into a searchable format with precise timestamps.**

---

**2. Program Overview**:

The program consists of the following components:

- **Audio Extraction**: The program extracts audio from a video file.

- **Speech Recognition**: Using the SpeechRecognition library, the audio content is transcribed into text.

- **Timestamp** Generation: Each word in the recognized text is associated with its occurrence timestamps.

- **Search Functionality:** Users can search for specific words to retrieve their timestamps.

---

**3. Libraries Used:**

- speech\_recognition: A library for performing speech recognition.

- moviepy.editor: A library for working with video files.

- re: A built-in library for regular expressions.

---

**4. Program Execution:**

```python

import speech\_recognition as sr

import moviepy.editor as mp

import re

**# Load the video file and extract its audio**

clip = mp.VideoFileClip(r"motivation.mp4")

clip.audio.write\_audiofile(r"converted\_mp3.wav")

**# Initialize the speech recognizer**

r = sr.Recognizer()

audio = sr.AudioFile(r"converted\_mp3.wav")

**# Open the audio file and record its conten**t

with audio as source:

audio\_file = r.record(source)

**# Recognize speech from the audio file**

result = r.recognize\_google(audio\_file)

**# Split the recognized text into individual words using regular expression**

words = re.findall(r'\b\w+\b', result.lower())

**# Store words and their timestamps in a dictionary**

word\_timestamps = {}

for i, word in enumerate(words):

word\_timestamps.setdefault(word, []).append(i)

**# Write the recognized speech and timestamps to a file**

with open('recog.txt', mode='w') as file:

file.write("Speech Recognized\n")

file.write(result)

file.write("\n\nWord Timestamps:\n")

for word, timestamps in word\_timestamps.items():

file.write(f"{word}: {timestamps}\n")

print("File is ready.")

**# Prompt the user to search for a word**

search\_word = input("Enter a word to search: ").lower()

**# Check if the word exists in the timestamps**

if search\_word in word\_timestamps:

print(f"'{search\_word}' found at timestamps: {word\_timestamps[search\_word]}")

else:

print(f"'{search\_word}' not found.")

print("Enter 'exit' to exit from the program.")

```

---

**5. Conclusion:**

The Speech to Text Converter with Timestamps provides a convenient way to convert spoken content into searchable text with precise timestamps. By leveraging libraries such as speech\_recognition and moviepy, the program offers efficient audio processing capabilities. Additionally, the inclusion of search functionality enhances the usability of the generated timestamps.

---

**6. Additional Information:**

- The program utilizes the Google Web Speech API for speech recognition. Ensure an active internet connection for optimal performance.

- For large video files, the audio extraction process may take longer durations.

- Users can customize the program to support different speech recognition engines or improve timestamp accuracy through advanced algorithms.

---

This documentation outlines the functionality, implementation, and usage of the Speech to Text Converter with Timestamps project. It serves as a comprehensive guide for understanding and utilizing the provided Python program.